MEMOTE for standardized genome-scale metabolic model testing

Citation for the original published paper (version of record):
Lieven, C., Beber, M., Olivier, B. et al (2020). MEMOTE for standardized genome-scale metabolic model testing. Nature Biotechnology, 38(3): 272-276. http://dx.doi.org/10.1038/s41587-020-0446-y

OPEN

MEMOTE for standardized genome-scale metabolic model testing

To the Editor — Reconstructing metabolic reaction networks enables the development of testable hypotheses of an organism's metabolism under different conditions[1]. State-of-the-art genome-scale metabolic models (GEMs) can include thousands of metabolites and reactions that are assigned to subcellular locations. Gene–protein–reaction (GPR) rules and annotations using database information can add meta-information to GEMs. GEMs with metadata can be built using standard reconstruction protocols[2], and guidelines have been put in place for tracking provenance and enabling interoperability, but a standardized means of quality control for GEMs is lacking[3]. Here we report a community effort to develop a test suite named MEMOTE (for metabolic model tests) to assess GEM quality.

Incompatible description formats and missing annotations[4] limit GEM reuse. Moreover, numerical errors[5] and omission of essential cofactors[6] in a single biomass objective function can have substantial impact on the predictive performance of a GEM. Failure to make checks for flux cycles and imbalances can render model predictions untrustworthy[7].

Every year, increasing numbers of manually curated and automatically generated GEMs are published, including those for human and cancer tissue models[8]. We believe that it is essential to optimize GEM reproducibility and reuse. Researchers need models that are software-agnostic, with components that have standardized, database-independent identifiers. Default conditions and mathematically specified modeling formulations must be precisely defined to allow reproduction of the original model predictions. Models must produce feasible phenotypes under various conditions. Finally, data used to build any model must be made available in a reusable format.

A dual approach could be used to improve GEM reuse and reproducibility. First, we advocate adoption of the latest version of the Systems Biology Markup Language (SBML) level 3 flux balance constraints (SBML3FBC) package[9] as the primary description and exchange format. The SBML3FBC package adds structured, semantic descriptions for domain-specific model components such as flux bounds, multiple linear objective functions, GPR rules, metabolite chemical formulas, charge and annotations. The SBML and constraint-based modeling communities collaboratively develop this package, updating it based on user input. It has been adopted by a wide range of constraint-based modeling software and public model repositories (http://cbmpy.sourceforge.net/ and refs. [10–15]), and should therefore be considered the standard for encoding GEMs.

Second, we present MEMOTE (/'mi:moʊt/ in international phonetic alphabet notation), an open-source Python software that represents a unified approach to ensure the formally correct definition of SBML3FBC and provides quality control and continuous quality assurance of metabolic models with tools and best practices already used in software development[16,17]. MEMOTE accepts stoichiometric models encoded in SBML3FBC and previous versions as input. In addition to structural validation analogous to the SBML validator[18], MEMOTE benchmarks metabolic models using consensus tests from four general areas: annotation, basic tests, biomass reaction and stoichiometry.

Annotation tests check that a model is annotated according to community standards with minimum information required in annotation of models (MIRIAM)-compliant cross-references[19], that all primary identifiers belong to the same namespace rather than being fractured across several namespaces, and that components are described using Systems Biology Ontology (SBO) terms[20]. A lack of explicit, standardized annotations complicates the use, comparison and extension of GEMs, and thus strongly hampers collaboration[3,4].

Basic tests check the formal correctness of a model and verify the presence of components such as metabolites, compartments, reactions and genes. These tests also check for metabolite formula and charge information, and GPR rules. General quality metrics, such as the degree of metabolic coverage representing the ratio of reactions and genes[21], are also checked.

A model is tested for production of biomass precursors in different conditions, for biomass consistency, for nonzero growth rate and for direct precursors. The biomass reaction is based on the biomass composition of the modeled organism and expresses its ability to produce the necessary precursors for in silico cell growth and maintenance. Thus, an extensive, well-formed biomass reaction is crucial for accurate predictions with a GEM[6].

Stoichiometric inconsistency, erroneously produced energy metabolites[7] and permanently blocked reactions are identified by MEMOTE. Errors in stoichiometries may result in the production of ATP or redox cofactors from nothing[2] and are detrimental to the performance of the model when using flux-based analysis[4].

MEMOTE enables a quick comparison of any two given models, in which individual test results are quantified and condensed to calculate an overall score (Supplementary Note 1). In addition to these consensus tests, researchers can supply experimental data from growth and gene perturbation studies in a range of input formats (.csv, .tsv, .xls or .xslx) in MEMOTE. To support reproducibility, researchers can configure MEMOTE to recognize specific data types as input to predefined experimental tests for model validation (Supplementary Note 2).

There are two main workflows for MEMOTE (Fig. 1a and Supplementary Figs. 1–3). For peer review, MEMOTE can produce either a 'snapshot report' or a 'diff report' that display MEMOTE test results of one single or multiple models, respectively. For model reconstruction, MEMOTE helps users to create a version-controlled repository of the model and to activate continuous integration toward building a 'history report' that records the results of each tracked edit of the model. Although a model repository can be used offline, we encourage community collaboration via distributed version control development platforms, such as GitHub (https://github.com), GitLab (https://gitlab.com/) or BioModels[12] (http://wwwdev.ebi.ac.uk/biomodels/). MEMOTE is tightly integrated with GitHub. Models generated and versioned in MEMOTE can easily be uploaded to GitLab and BioModels. Collaborative model reconstruction with MEMOTE as benchmark can occur using all three software platforms (Fig. 1b).

We validated MEMOTE using models from seven GEM collections (Fig. 2, Supplementary Table 1 and Supplementary Methods), that comprise manually and (semi)-automatically reconstructed GEMs (10,780 models in total). Most GEM collections have already made models available in SMBL format. A nonlinear dimensional reduction of the normalized test results (Supplementary Methods) using *t*-distributed stochastic neighbor embedding

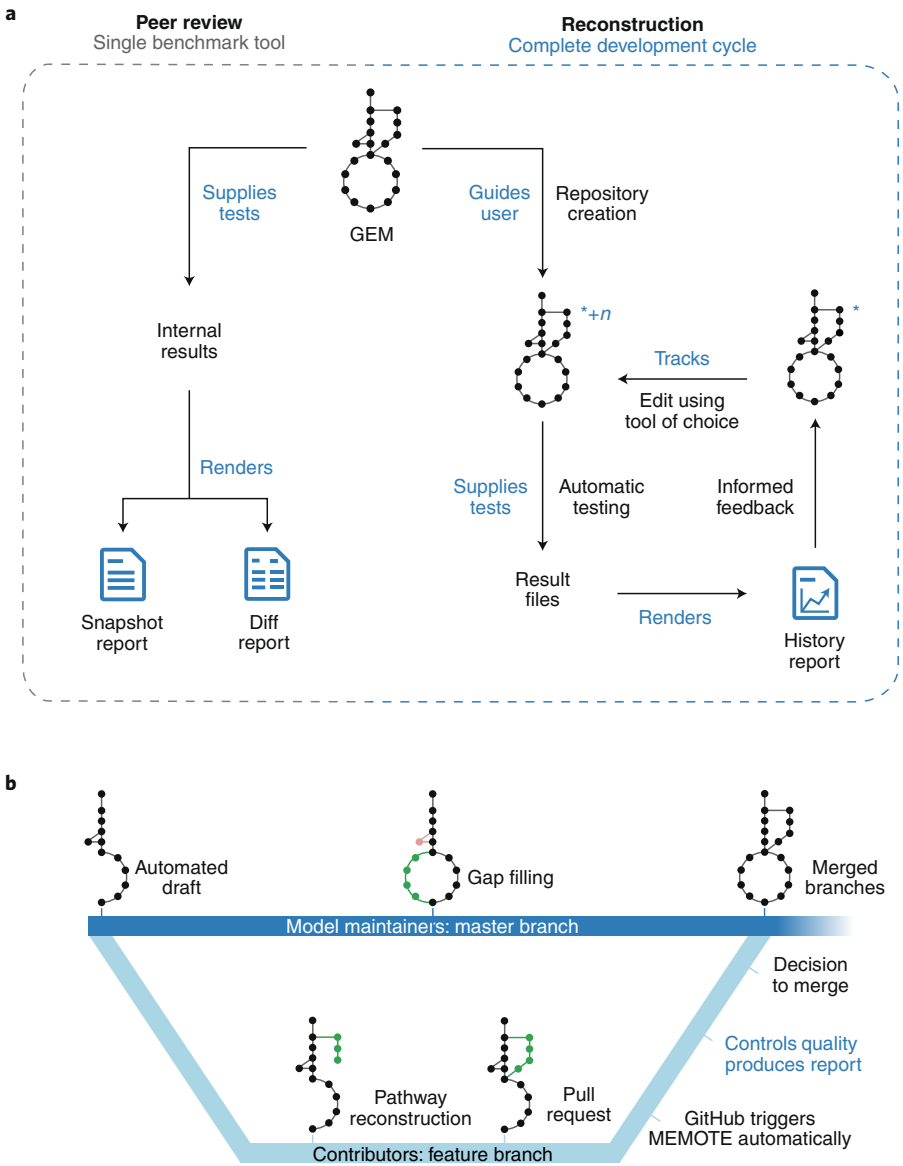

Fig. 1 | Graphical summary of MEMOTE. a, Graphical representation of the two principal workflows in detail. For peer review, MEMOTE serves as a benchmark tool generating a comprehensive, human-readable report, which quantifies the model's performance (Supplementary Figs. 1 and 2). With this information, a definitive assessment of model quality can be made by editors, reviewers and users. This workflow is accessible through a web interface (https://memote.io) or locally through a command line interface. For model reconstruction, MEMOTE helps users to create a version-controlled repository for the model (indicated by the blue asterisk), and to activate continuous integration. The model is tested using MEMOTE's library of test cases, the results are saved, and an initial report of the model is generated. This constitutes the first iteration of the development cycle. Now, users may edit the model using their preferred reconstruction tool and subsequently export it to SBML3FBC, thus creating a new version (indicated by +*n*). This will restart the cycle by running the tests automatically, saving the results for each version and including them incrementally in a report on the entire history of results. This serves as a guide toward a functional, high-quality GEM (Supplementary Fig. 3). This workflow is accessible through the command line only. **b**, Both, GitHub and GitLab support a branching strategy, which model builders could use to curate different parts of the model simultaneously or to invite external experts to improve specific model features. MEMOTE further enables model authors to act as gatekeepers, choosing to accept only high-quality contributions. Identification of functional differences happens in the form of a comparative 'diff' report, whereas for file-based discrepancies MEMOTE capitalizes on the platform's ability to show the line-by-line changes between different versions of a model. For this purpose, the model is written in a sorted YAML format[28] after every change. Bold blue text denotes actions performed by MEMOTE.

(*t*-SNE; Fig. 2a) indicates that models from the same source are generally more similar to each other than to models from other sources. Nevertheless, several model sources reveal internal subgroupings (Fig. 2a). With the exception of Path2Models[22], which relies on pathway resources that contain problematic reaction information on stoichiometry and directionality[23], automatically reconstructed GEMs were stoichiometrically consistent (Fig. 2b) and mass-balanced (Supplementary Fig. 4). Of the manually reconstructed GEMs we tested, most models in BiGG[13] are stoichiometrically consistent, but there is wide variation among published models, with ~70% of models having at least one stoichiometrically unbalanced metabolite. Stoichiometrically inconsistent models cannot be mass-balanced, but missing formula annotations, from which molecular masses are calculated, further contribute to reactions being counted as unbalanced. The problems that we identified in published models underpin the need for application of MEMOTE during peer-review process (but ideally before submission) of GEMs.

During GEM reconstruction, metabolic reactions are defined based on functional gene annotations, and this information is output as GPR rules. We found that ~15% of reactions in models we tested are not annotated with GPR rules (Fig. 2c). For published models, subgroups of models contain up to 85% of reactions without GPR rules. This could be due to a large number of modeling-specific reactions, spontaneous reactions[24] and known reactions with undiscovered genes, or if GPR rules were annotated in nonstandard ways.

CarveMe[25] and Path2Models[22] have a very low fraction of universally blocked reactions, whereas models from AGORA[26] and KBase[14] contain ~30% blocked reactions, and BiGG[13] models and OptFlux[15] models contain ~20% blocked reactions (Fig. 2d). Similarly, orphan and dead-end metabolites (Supplementary Figs. 5 and 6) are also present in all of these published collections. We note that blocked reactions and dead-end metabolites are not indicators of low-quality models but that a large proportion (for example, >50%) of universally blocked reactions can indicate problems in reconstruction that need solving.

AGORA, KBase and BiGG are the only collections with SBML-compliant metabolite and reaction annotations. Gene annotations are only present in KBase models and selected BiGG models (Supplementary Figs. 7–9). Each collection uses its own system of identifiers for each model component, but there is some overlap between all three (Supplementary Figs. 10

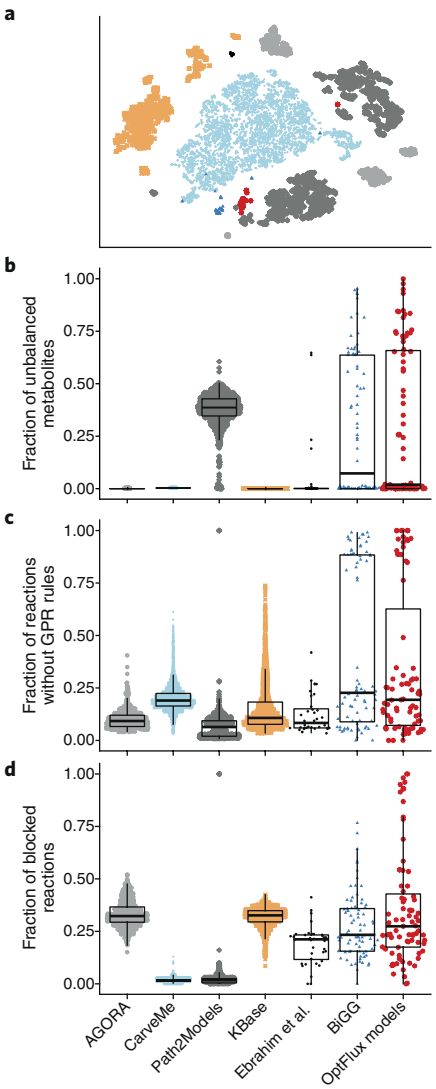

Fig. 2 | Quality of manually reconstructed GEMs from collections without quality control or quality assurance. a, Depicted is a *t*-SNE two-dimensional reduction of models using normalized test features as input. Only GEMs from the BiGG collection form a single albeit small cluster. Models from all other collections are grouped in several fragmented but distinct clusters. **b**–**d**, SinaPlots[29] of each collection overlaid with box and whisker plots to indicate 25%, 50% (median) and 75% quantiles. GEMs from collections built in a modern automated pipeline (AGORA, CarveMe, KBase) are stoichiometrically consistent, whereas models from the older Path2Models collection are up to 50% stoichiometrically inconsistent (**b**). Manually reconstructed models (BiGG, Ebrahim et al.[30], OptFlux models) contain varying degrees of inconsistent GEMs. GPR rules are essential for in silico knockout studies, but also serve to justify the presence of a reaction (**c**). Generally, the fraction of reactions without GPR rules is low (~15%). Yet a distinct group of models from the collections of Ebrahim et al. and OptFlux lack GPR rules for >75% of their reactions. Most models from the CarveMe and Path2Models collections contain very few blocked reactions, whereas for models from the other collections the number of blocked reactions lies mostly between 10% and 30% (**d**). Again, models from the collections of Ebrahim et al. and the Optflux models show the largest variance.

and 11), and partial overlaps for models from KBase and BiGG (Supplementary Figs. 12–16), or AGORA and BiGG (Supplementary Figs. 17 and 18), but not KBase and AGORA. BiGG is the only collection with models using MetaNetX[27] annotations (Supplementary Fig. 19). MetaNetX consolidates biochemical namespaces by establishing a mapping between them through a set of unique identifiers. Hence, knowing the MetaNetX identifier for a given entity often means also knowing the identifiers for other databases (Supplementary Methods).

MEMOTE tests cover semantic and conceptual requirements, which are fundamental to SBML3FBC and constraint-based modeling, respectively. They are extensible to allow the validation of a model's performance against experimental data and can be executed as a stand-alone tool or integrated into existing reconstruction pipelines. Capitalizing on robust workflows established in modern software development, MEMOTE promotes openness and collaboration by granting the community tangible metrics to support their research and to discuss assumptions or limitations openly.

Application of a set of defined metabolic model tests is not dependent on implementation in MEMOTE, and for some users it may be more desirable to implement each test separately to streamline the user experience.

We propose that an independent, central library of tests and a tool to run them offers an unbiased approach to quality control because the tests are continuously reviewed by the community. This resource will be maintained under stewardship of Nikolaus Sonnenschein by the openCOBRA consortium (https://github.com/opencobra). To encourage integration as opposed to duplication, MEMOTE provides a Python application programing interface (API) as well as being available as a web service. MEMOTE has already been integrated in several services and tools (Supplementary Note 3). We discuss alternatives and future perspectives of MEMOTE in Supplementary Notes 4 and 5, respectively.

We recommend that MEMOTE users reach out to GEM authors to report any errors and thereby enable community improvement of models as resources. Using inconsistent GEMs for hypothesis generation could lead researchers down blind alleys, so we weighed the influence of 'consistency' and 'stoichiometric consistency' and SBO terms higher than tests for metabolite, reaction and gene annotations.

We are committed to keeping MEMOTE open to support community principles. Robust benchmarking will only work if it is actively supported by the whole community, and we call on any interested experts to join this endeavor and enable its continual improvement.

Reporting Summary

Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Data availability

The model collection is available at https://doi.org/10.5281/zenodo.2636858. Individual results and aggregated tables, as well as analysis code, are available at https://doi.org/10.5281/zenodo.2638234.

Code availability

MEMOTE source code is available at https://github.com/opencobra/memote under the Apache license, version 2.0. Supporting documentation is available at https://memote.readthedocs.io/en/latest/. The MEMOTE web interface is hosted at https://memote.io. A detailed list of all tests in MEMOTE is available at https://memote.readthedocs.io/en/latest/autoapi/index.html. ❐

Christian Lieven[1,41], Moritz E. Beber[1,41], Brett G. Olivier[2], Frank T. Bergmann[3], Meric Ataman[4], Parizad Babaei[1], Jennifer A. Bartell[1], Lars M. Blank[5], Siddharth Chauhan[6], Kevin Correia[7], Christian Diener[8,9], Andreas Dräger[10,11,12], Birgitta E. Ebert[5,13], Janaka N. Edirisinghe[14], José P. Faria[14], Adam M. Feist[1,6], Georgios Fengos[4], Ronan M. T. Fleming[15], Beatriz García-Jiménez[16,40], Vassily Hatzimanikatis[4], Wout van Helvoirt[17,18], Christopher S. Henry[14], Henning Hermjakob[19], Markus J. Herrgård[1], Ali Kaafarani[1], Hyun Uk Kim[20], Zachary King[6], Steffen Klamt[21], Edda Klipp[22], Jasper J. Koehorst[23],

Matthias König[22], Meiyappan Lakshmanan[24], Dong-Yup Lee[24,25], Sang Yup Lee[1,20], Sunjae Lee[26,27], Nathan E. Lewis[6,28], Filipe Liu[14], Hongwu Ma[29], Daniel Machado[30], Radhakrishnan Mahadevan[7,31], Paulo Maia[32], Adil Mardinoglu[26,27], Gregory L. Medlock[33], Jonathan M. Monk[6], Jens Nielsen[1,34], Lars Keld Nielsen[1,13], Juan Nogales[16], Intawat Nookaew[34,35], Bernhard O. Palsson[1,6], Jason A. Papin[33], Kiran R. Patil[30], Mark Poolman[36], Nathan D. Price[9], Osbaldo Resendis-Antonio[8], Anne Richelle[28], Isabel Rocha[37,38], Benjamín J. Sánchez[1,34], Peter J. Schaap[23], Rahuman S. Malik Sheriff[19], Saeed Shoaie[26,27], Nikolaus Sonnenschein[1]✉, Bas Teusink[2], Paulo Vilaça[32], Jon Olav Vik[17], Judith A. H. Wodke[22], Joana C. Xavier[39], Qianqian Yuan[29], Maksim Zakhartsev[17] and Cheng Zhang[26]

*[1]Novo Nordisk Foundation Center for Biosustainability, Technical University of Denmark, Lyngby, Denmark. [2]Systems Biology Lab, Amsterdam Institute of Molecular and Life Sciences (AIMMS), Vrije Universiteit Amsterdam, Amsterdam, the Netherlands. [3]BioQUANT/COS, Heidelberg University, Heidelberg, Germany. [4]Ecole Polytechnique Fédérale de Lausanne, Laboratory of Computational Systems Biotechnology, Lausanne, Switzerland. [5]iAMB-Institute of Applied Microbiology, ABBt-Aachen Biology and Biotechnology, RWTH Aachen University, Aachen, Germany. [6]Department of Bioengineering, University of California, La Jolla, CA, USA. [7]Department of Chemical Engineering and Applied Chemistry, University of Toronto, Toronto, Ontario, Canada. [8]Human Systems Biology Laboratory, Instituto Nacional de Medicina Genomica & Coordinación de la Investigación Científica-Red de Apoyo a la Investigación, UNAM, Mexico City, Mexico. [9]Institute for Systems Biology, Seattle, WA, USA. [10]Computational Systems Biology of Infection and Antimicrobial-Resistant Pathogens, Institute for Biomedical Informatics (IBMI), Tübingen, Germany. [11]Department of Computer Science, University of Tübingen, Tübingen, Germany. [12]German Center for Infection Research (DZIF), partner site Tübingen, Tübingen, Germany. [13]Australian Institute for Bioengineering and Nanotechnology, The University of Queensland, Brisbane, Queensland, Australia. [14]Argonne National Laboratory, Lemont, IL, USA. [15]Analytical Biosciences, Division of Systems Biomedicine and Pharmacology, Leiden Academic Centre for Drug Research, Leiden University, Leiden, the Netherlands. [16]Department of Systems Biology, Centro Nacional de Biotecnología, Consejo Superior de Investigaciones Científicas (CNB-CSIC), Madrid, Spain. [17]Department of Animal and Aquacultural Sciences, Faculty of Biosciences, Norwegian University of Life Sciences, Oslo, Norway. [18]Hanze University of Applied Sciences, Groningen, the Netherlands. [19]European Bioinformatics Institute, European Molecular Biology Laboratory (EMBL-EBI), Wellcome Trust Genome Campus, Cambridge, UK. [20]Department of Chemical and Biomolecular Engineering (BK21 Plus Program), BioProcess Engineering Research Center, BioInformatics Research Center, Institute for the BioCentury, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea. [21]Analysis and Redesign of Biological Networks, Max Planck Institute for Dynamics of Complex Technical Systems Magdeburg, Magdeburg, Germany. [22]Theoretical Biophysics, Humboldt-Universität zu Berlin, Berlin, Germany. [23]Department of Agrotechnology and Food Sciences, Laboratory of Systems and Synthetic Biology, Wageningen University & Research, Wageningen, the Netherlands. [24]Bioprocessing Technology Institute, Agency for Science, Technology and Research (A*STAR), Singapore, Singapore. [25]School of Chemical Engineering Sungkyunkwan University, Jangan-gu Suwon, Gyeonggi-do, Republic of Korea. [26]Science for Life Laboratory, KTH - Royal Institute of Technology, Stockholm, Sweden. [27]Centre for Host-Microbiome Interactions, Faculty of Dentistry, Oral & Craniofacial Sciences, King's College London, London, UK. [28]Department of Pediatrics and Novo Nordisk Foundation Center for Biosustainability, University of California, San Diego School of Medicine, La Jolla, CA, USA. [29]Key Laboratory of Systems Microbial Biotechnology, Tianjin Institute of Industrial Biotechnology, Chinese Academy of Sciences, Tianjin, P.R. China. [30]Structural and Computational Biology Unit, European Molecular Biology Laboratory (EMBL), Heidelberg, Germany. [31]Institute of Biomaterials and Biomedical Engineering, University of Toronto, Toronto, Canada. [32]SilicoLife Lda., Braga, Portugal. [33]Department of Biomedical Engineering, University of Virginia, Charlottesville, Virginia, USA. [34]Chalmers University of Technology, Department of Biology and Biological Engineering, Division of Systems and Synthetic Biology, Göteborg, Sweden. [35]Department of Biomedical Informatics, College of Medicine, University of Arkansas for Medical Sciences (UAMS), Little Rock, AR, USA. [36]Oxford Brookes University, Oxford, UK. [37]Centre of Biological Engineering, University of Minho, Braga, Portugal. [38]Instituto de Tecnologia Química e Biológica António Xavier, Universidade Nova de Lisboa (ITQB-NOVA), Oeiras, Portugal. [39]Institute for Molecular Evolution, Heinrich-Heine-University Düsseldorf, Düsseldorf, Germany. [40]Present address: Centro de Biotecnología y Genómica de Plantas (CBGP, UPM-INIA), Universidad Politécnica de Madrid (UPM) – Instituto Nacional de Investigación y Tecnología Agraria y Alimentaria (INIA), Madrid, Spain. [41]These authors contributed equally: Christian Lieven, Moritz E. Beber.*
✉*e-mail: niso@biosustain.dtu.dk*

Published online: 2 March 2020
https://doi.org/10.1038/s41587-020-0446-y

References

1. Palsson, B.Ø. *Systems Biology: Constraint-based Reconstruction and Analysis* (Cambridge Univ. Press, 2015).
2. Thiele, I. & Palsson, B. Ø. *Nat. Protoc.* **5**, 93–121 (2010).
3. Heavner, B. D. & Price, N. D. *Curr. Opin. Biotechnol.* **34**, 105–109 (2015).
4. Ravikrishnan, A. & Raman, K. *Brief. Bioinform.* **16**, 1057–1068 (2015).
5. Chan, S.H.J., Cai, J., Wang, L., Simons-Senftle, M.N. & Maranas, C.D. *Bioinformatics* https://doi.org/10.1093/bioinformatics/btx453 (2017).
6. Xavier, J. C., Patil, K. R. & Rocha, I. *Metab. Eng.* **39**, 200–208 (2017).
7. Fritzemeier, C. J., Hartleb, D., Szappanos, B., Papp, B. & Lercher, M. J. *PLoS Comput. Biol.* **13**, e1005494 (2017).
8. Jerby, L. & Ruppin, E. *Clin. Cancer Res.* **18**, 5572–5584 (2012).
9. Olivier, B.G. & Bergmann, F.T. *J. Integr. Bioinform.* **15**, 20170082 (2018).
10. Heirendt, L. et al. *Nat. Protoc.* **14**, 639–702 (2019).
11. Ebrahim, A., Lerman, J. A., Palsson, B. O. & Hyduke, D. R. *BMC Syst. Biol.* **7**, 74 (2013).
12. Chelliah, V. et al. *Nucleic Acids Res.* **43**, D542–D548 (2015).
13. King, Z. A. et al. *Nucleic Acids Res.* **44**, D515–D522 (2016).
14. Arkin, A. P. et al. *Nat. Biotechnol.* **36**, 566–569 (2018).
15. Rocha, I. et al. *BMC Syst. Biol.* **4**, 45 (2010).
16. Cooper, J., Vik, J. O. & Waltemath, D. *Prog. Biophys. Mol. Biol.* **117**, 99–106 (2015).
17. Beaulieu-Jones, B. K. & Greene, C. S. *Nat. Biotechnol.* **35**, 342–346 (2017).
18. Bornstein, B. J., Keating, S. M., Jouraku, A. & Hucka, M. *Bioinformatics* **24**, 880–881 (2008).
19. Le Novère, N. et al. *Nat. Biotechnol.* **23**, 1509–1515 (2005).
20. Courtot, M. et al. *Mol. Syst. Biol.* **7**, 543 (2011).
21. Monk, J., Nogales, J. & Palsson, B. O. *Nat. Biotechnol.* **32**, 447–452 (2014).
22. Büchel, F. et al. *BMC Syst. Biol.* **7**, 116 (2013).
23. Yuan, Q. et al. *PLoS One* **12**, e0169437 (2017).
24. Keller, M. A., Piedrafita, G. & Ralser, M. *Curr. Opin. Biotechnol.* **34**, 153–161 (2015).
25. Machado, D., Andrejev, S., Tramontano, M. & Patil, K. R. *Nucleic Acids Res.* **46**, 7542–7553 (2018).
26. Magnúsdóttir, S. et al. *Nat. Biotechnol.* **35**, 81–89 (2017).
27. Moretti, S. et al. *Nucleic Acids Res.* **44**, D523–D526 (2016).
28. Steffensen, J. L., Dufault-Thompson, K., Zhang, Y. & Dandekar, T. PSAMM: a portable system for the analysis of metabolic models. *PLOS Comput. Biol.* **12**, e1004732 (2016).
29. Sidiropoulos, N. et al. SinaPlot: an enhanced chart for simple and truthful representation of single observations over multiple classes. *J. Comput. Graph. Stat.* **27**, 673–676 (2018).
30. Ebrahim, A. et al. Do genome-scale models need exact solvers or clearer standards? *Mol. Syst. Biol.* **11**, 831 (2015).

Acknowledgements

We acknowledge D. Dannaher and A. Lopez for their supporting work on the Angular parts of MEMOTE; resources and support from the DTU Computing Center; J. Cardoso, S. Gudmundsson, K. Jensen and D. Lappa for their feedback on conceptual details; and P. D. Karp and I. Thiele for critically reviewing the manuscript. We thank J. Daniel, T. Kristjánsdóttir, J. Saez-Saez, S. Sulheim, and P. Tubergen for being early adopters of MEMOTE and for providing written testimonials. J.O.V. received the Research Council of Norway grants 244164 (GenoSysFat), 248792 (DigiSal) and 248810 (Digital Life Norway); M.Z. received the Research Council of Norway grant 244164 (GenoSysFat); C.L. received funding from the Innovation Fund Denmark (project "Environmentally Friendly Protein Production (EFPro2)"); C.L., A.K., N. S., M.B., M.A., D.M., P.M, B.J.S., P.V., K.R.P. and M.H. received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement 686070 (DD-DeCaF); B.G.O., F.T.B. and A.D. acknowledge funding from the US National Institutes of Health (NIH, grant number 2R01GM070923-13); A.D. was supported by infrastructural funding from the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), Cluster of Excellence EXC 2124 Controlling Microbes to Fight Infections; N.E.L. received funding from NIGMS R35 GM119850, Novo Nordisk Foundation NNF10CC1016517 and the Keck Foundation; A.R. received a Lilly Innovation Fellowship Award; B.G.-J. and J. Nogales received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no 686585 for the project LIAR, and the Spanish Ministry of Economy

and Competitivity through the RobDcode grant (BIO2014-59528-JIN); L.M.B. has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement 633962 for project P4SB; R.F. received funding from the US Department of Energy, Offices of Advanced Scientific Computing Research and the Biological and Environmental Research as part of the Scientific Discovery Through Advanced Computing program, grant DE-SC0010429; A.M., C.Z., S.L. and J. Nielsen received funding from The Knut and Alice Wallenberg Foundation, Advanced Computing program, grant #DE-SC0010429; S.K.'s work was in part supported by the German Federal Ministry of Education and Research (de.NBI partner project "ModSim" (FKZ: 031L104B)); E.K. and J.A.H.W. were supported by the German Federal Ministry of Education and Research (project "SysToxChip", FKZ 031A303A); M.K. is supported by the Federal Ministry of Education and Research (BMBF, Germany) within the research network Systems Medicine of the Liver (LiSyM, grant number 031L0054); J.A.P. and G.L.M. acknowledge funding from US National Institutes of Health (T32-LM012416, R01-AT010253, R01-GM108501) and the Wagner Foundation; G.L.M. acknowledges funding from a Grand Challenges Exploration Phase I grant (OPP1211869) from the Bill & Melinda Gates Foundation; H.H. and R.S.M.S. received funding from the Biotechnology and Biological Sciences Research Council MultiMod (BB/N019482/1); H.U.K. and S.Y.L. received funding from the Technology Development Program to Solve Climate Changes on Systems Metabolic Engineering for Biorefineries (grants NRF-2012M1A2A2026556 and NRF-2012M1A2A2026557) from the Ministry of Science and ICT through the National Research Foundation (NRF) of Korea; H.U.K. received funding from the Bio & Medical Technology Development Program of the NRF, the Ministry of Science and ICT (NRF-2018M3A9H3020459); P.B., B.J.S., Z.K., B.O.P., C.L., M.B., N.S., M.H. and A.F. received funding through Novo Nordisk Foundation through the Center for Biosustainability at the Technical University of Denmark (NNF10CC1016517); D.-Y.L. received funding from the Next-Generation BioGreen 21 Program (SSAC, PJ01334605), Rural Development Administration, Republic of Korea; G.F. was supported by the RobustYeast within ERA net project via SystemsX.ch; V.H. received funding from the ETH Domain and Swiss National Science Foundation; M.P. acknowledges Oxford Brookes University; J.C.X. received support via European Research Council (666053) to W.F. Martin; B.E.E. acknowledges funding through the CSIRO-UQ Synthetic Biology Alliance; C.D. is supported by a Washington Research Foundation Distinguished Investigator Award. I.N. received funding from National Institutes of Health (NIH)/National Institute of General Medical Sciences (NIGMS) (grant P20GM125503).

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information is available for this paper at https://doi.org/10.1038/s41587-020-0446-y.

The nf-core framework for community-curated bioinformatics pipelines

To the Editor — The standardization, portability and reproducibility of analysis pipelines are key issues within the bioinformatics community. Most bioinformatics pipelines are designed for use on-premises; as a result, the associated software dependencies and execution logic are likely to be tightly coupled with proprietary computing environments. This can make it difficult or even impossible for others to reproduce the ensuing results, which is a fundamental requirement for the validation of scientific findings. Here, we introduce the nf-core framework as a means for the development of collaborative, peer-reviewed, best-practice analysis pipelines (Fig. 1). All nf-core pipelines are written in Nextflow and so inherit the ability to be executed on most computational infrastructures, as well as having native support for container technologies such as Docker and Singularity. The nf-core community (Supplementary Fig. 1) has developed a suite of tools that automate pipeline creation, testing, deployment and synchronization. Our goal is to provide a framework for high-quality bioinformatics pipelines that can be used across all institutions and research facilities.

Being able to reproduce scientific results is the central tenet of the scientific method. However, moving toward FAIR (findable, accessible, interoperable and reusable) research methods[1] in data-driven science is complex[2,3]. Central repositories, such as bio.tools[4], omictools[5] and the Galaxy toolshed[6], make it possible to find existing pipelines and their associated tools. However, it is still notoriously challenging to develop analysis pipelines that are fully reproducible and interoperable across multiple systems and institutions — primarily because of differences in hardware, operating systems and software versions.

Although the recommended guidelines for some analysis pipelines have become standardized (for example, GATK best practices[7]), the actual implementations are usually developed on a case-by-case basis. As such, there is often little incentive to test, document and implement pipelines in a way that permits their reuse by other researchers. This can hamper sustainable sharing of data and tools, and results in a proliferation of heterogeneous analysis pipelines, making it difficult for newcomers to find what they need to address a specific analysis question.

As the scale of -omics data and their associated analytical tools has grown, the scientific community is increasingly moving toward the use of specialized workflow management systems to build analysis pipelines[8]. They separate the requirements of the underlying compute infrastructure from the analysis and workflow description, introducing a higher degree of portability as compared to custom in-house scripts. One such popular tool is Nextflow[9]. Using Nextflow, software packages can be bundled with analysis pipelines using built-in integration for package managers, such as Conda, and containerization platforms, such as Docker and Singularity. Moreover, support for most common high-performance-computing batch schedulers and cloud providers allows simple deployment of analysis pipelines on almost any infrastructure. The opportunity to run pipelines locally during initial development and then to proceed seamlessly to large-scale computational resources in high-performance-computing or cloud settings provides users and developers with great flexibility.

The nf-core community project collects a curated set of best-practice analysis pipelines built using Nextflow. Similar projects include the 'awesome-pipelines' repository, which provides an extensive list of pipelines developed by the Nextflow community (https://github.com/pditommaso/awesome-pipeline), although these pipelines are variable in terms of development status and design. High-level approaches to facilitate the creation of end-to-end analysis pipelines are also available: Flowcraft (https://github.com/assemblerflow/flowcraft) and Pipeliner[10] are

nature research

Corresponding author(s): Nikolaus Sonnenschein

Last updated by author(s): Jan 28, 2020

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see Authors & Referees and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☒	☐	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☒	☐	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐	A description of all covariates tested
☒	☐	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☒	☐	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☒	☐	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	Data (models) was download from publicly available websites.
Data analysis	All code relevant to the manuscript is available at https://github.com/opencobra/memote, https://github.com/opencobra/memote-webservice, https://github.com/opencobra/cookiecutter-memote, https://github.com/opencobra/memote-docker, and https://github.com/biosustain/memote-meta-study/. At submission of the revised manuscript, memote had version 0.9.6 and cobrapy version 0.14.2 (the versions of all software dependencies).

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors/reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

The collected data (metabolic models) and code behind Figure 4 are available at https://github.com/biosustain/memote-meta-study/ and furthermore deposited to zenodo.org (https://doi.org/10.5281/zenodo.2638233)

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	We aimed for the broadest possible assessment of publicly available genome-scale metabolic models. Sample size was therefore determined by the availability of data.
Data exclusions	Models that could not be parsed by memote because they were either not available in SBML or because the SBML files were invalid have been excluded from the study.
Replication	Not applicable.
Randomization	Not applicable.
Blinding	Not applicable.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology
☒	☐ Animals and other organisms
☒	☐ Human research participants
☒	☐ Clinical data

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging